



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/609,395	06/27/2003	David H. Hanes	100201461-1	9366

22879 7590 01/25/2006

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER

WANG, DIANA S

ART UNIT PAPER NUMBER

2115

DATE MAILED: 01/25/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/609,395	Applicant(s) HANES, DAVID H.	
	Examiner Diana S. Wang	Art Unit 2115	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 27 June 2003.
- 2a) ☐ This action is FINAL. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-22 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-22 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☒ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>07/30/2003</u> . | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. Claims 1-22 are pending in this application.

Drawings

2. Figure 1 is a simplified schematic illustrative of an ASPI layer as may be conventionally implemented. It should be designated by a legend such as --Prior Art-- because only that which is old is illustrated. See MPEP § 608.02(g). Corrected drawings in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. The replacement sheet(s) should be labeled "Replacement Sheet" in the page header (as per 37 CFR 1.84(c)) so as not to obstruct any portion of the drawing figures. If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1 – 3 and 5 – 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent Application US 6,760,722 B1 issued to Hulikunta Prahlad

Art Unit: 2115

Raghunandan, Karnataka (IN) ("Raghunandan") in view of the publication "ASPI for Win32" issued by Adaptec, Inc. ("Adaptec").

With respect to claim 1, Raghunandan teaches a computer implemented system that receives an input/output requests (Figure 1, element 1, col 1, lines 49 – 51, col 6, lines 33 – 35); generates an input/output request formatted in accordance with a prescribed format (col 6, lines 33 – 34); and submits the generated input/output request to search engine for processing (Figure 1, element 5).

Raghunandan does not explicitly teach that input/output requests formatted in accordance with an application programming interface of an adapter interface layer as claimed.

Adaptec teaches claimed application programming interface of the adapter interface layer (Adaptec: page 1, section [Introduction] "ASPI allows developers to write device drivers and applications which are independent of the specific host adapter platform used to connect SCSI peripherals. This specification defines the protocol between ASPI client applications and the ASPI Manager", page 2, section [Using ASPI for Win 32 – An Overview], "ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests").

Raghunandan and Adaptec are analogous because both are directed for processing input/output requests (Raghunandan: col 1, lines 49 – 52, Adaptec: [An Overview]).

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Raghunandan and Adaptec. The combination of teachings would allow developers to write device drivers and applications which are independent of the specific host adapter platform.

With respect to claim 2, Raghunandan teaches to parse one or more field values of the received input/output request (Raghunandan: Figure 1, element 3, col 6, lines 40 – 41); and inserting the parsed values into respective fields of a prescribed format (Raghunandan: col 6, lines 49 – 51). Adaptec teaches the application programming interface of an adapter interface layer (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”).

As to claim 3, Adaptec teaches a request block data structure (Adaptec: page 4, section [SendASPI32Command Function], a SCSI Request Block (SRB) structure) formatted according to the adapter interface layer (Adaptec: page 5, section [Win32 ASPI Commands], the parsed member values of a request block is listed in Table 4 – 2).

As to claim 5, Adaptec teaches to identify a command type of the received input/output request (Adaptec: page 4 , “A command code is used to specify the type of I/O requested”).

As to claim 6, Adaptec teaches to identify a command type of the received input/output request and to generate input/output requests further comprises assigning a control code corresponding to the identified command type to an argument of the

generated input/output request (Adaptec: page 4 , "the command code is specified in a SCSI Request Block (SRB) structure. While SRB definition can vary depending on the ASPI command code, all SRBs include a standard SRB header" this is where command code is defined).

As to claim 7, Adaptec teaches to convey the generated input/output request to a host adapter (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], "ASPI function calls can be used to retrieve information about installed SCSI host adapters and devices", where SCSI host adapters and devices is a type of host adapter).

As to claim 8, Adaptec teaches to convey the generated input/output request to a small computer system interface host adapter (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], "ASPI function calls can be used to retrieve information about installed SCSI host adapters and devices").

As to claim 9, Adaptec teaches to receive a return data set formatted in accordance with the adapter interface layer (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], "ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests"); and translate the return data set into a format compatible with the application programming interface (Adaptec: page 3, Table 4 – 1 describes return values format, pages 5 – 18, Tables 4 – 3, 4 – 5, 4 – 7, 4 – 9, 4 – 11, 4 – 13 list possible return values).

As to claim 10, Adaptec teaches that the translated return data set is conveyed to a client application that generated the received input/output request (Adaptec: page 1, section [Introduction] "ASPI allows developers to write device drivers and applications

which are independent of the specific host adapter platform used to connect SCSI peripherals. This specification defines the protocol between ASPI client applications and the ASPI Manager”, page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”, page 3, Table 4 – 1 describes return values format, pages 5 – 18, Tables 4 – 3, 4 – 5, 4 – 7, 4 – 9, 4 – 11, 4 – 13 list possible return values).

5. Claims 4, 11 – 15 and 16 – 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent Application US 6,760,722 B1 issued to Hulikunta Prahlad Raghunandan, Karnataka (IN) (“Raghunandan”) in view of the publication “ASPI for Win32” issued by Adaptec, Inc. (“Adaptec”) and further in view of US Patent Application US 6,065,096 issued to Brian A. Day, et al (“Day”).

With respect to claim 4, Raghunandan teaches a computer implemented system that receives an input/output requests and generates an input/output request formatted in accordance with a prescribed format.

Raghunandan does not explicitly teach that input/output requests be formatted in accordance with an adapter interface layer as claimed.

Adaptec teaches to generate an input/output request formatted according to an adapter interface layer (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”), as claimed.

Raghunandan and Adaptec do not explicitly teach to generate the input/output request formatted according to a pass-through interface of an operating system, and submitting the generated input/output request to the pass-through interface.

Day teaches the claimed pass-through mode of operation that supports generating the input/output request and submitting the generated input/output request to the pass-through interface (Day: Figure 2, page 3, col 3, lines 10 – 25).

Raghunandan, Adaptec and Day are analogous because they all teach generating and submitting input/output requests.

With respect to claim 4, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Raghunandan, Adaptec and Day. The combination of teachings would allow applications to continue use existing application programming interfaces without changes to applications and drivers when operate in a compatible and pass-through mode.

With respect to claim 11, Raghunandan teaches a computer implemented system that receives an input/output requests (Raghunandan : Figure 1, element 1, col 1, lines 49 – 51, col 6, lines 33 – 35); generates an input/output request formatted in accordance with a prescribed format (Raghunandan : col 6, lines 33 – 34); and submits the generated input/output request to search engine for processing (Raghunandan : Figure 1, element 5).

Raghunandan does not explicitly teach that input/output request formatted in accordance with an application programming interface as claimed.

Adaptec teaches the claimed claimed application programming interface of the adapter interface layer (Adaptec: page 1, section [Introduction] "ASPI allows developers to write device drivers and applications which are independent of the specific host adapter platform used to connect SCSI peripherals. This specification defines the protocol between ASPI client applications and the ASPI Manager", page 2, section [Using ASPI for Win 32 – An Overview], "ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests").

Raghunandan and Adaptec do not explicitly teach to a translation layer, as claimed.

Day teaches a pass-through mode of operation that supports generating the input/output request and submitting the generated input/output request to the SCSI pass-through interface (Day: Figure 2, page 3, col 3, lines 10 – 25). The **SCSI Pass-through mode that can be used as a translation layer**, as claimed.

Raghunandan , Adaptec and Day are analogous because they all teach generating and submitting input/output requests.

With respect to claim 11, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Raghunandan, Adaptec and Day. The combination of teachings would allow applications to continue use existing application programming interfaces without changes to applications and drivers when operate in a compatible and pass-through mode.

Art Unit: 2115

As to claim 12, Adaptec teaches to convey, by the adapter interface layer, the generated input/output request to a host adapter (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”).

As to claim 13, Adaptec teaches to receive, by the adapter interface layer, a return data set formatted in accordance with the adapter interface layer (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”); and translate the return data set into a format compatible with the application programming interface (Adaptec: page 3, Table 4 – 1 describes return values format, pages 5 – 18, Tables 4 – 3, 4 – 5, 4 – 7, 4 – 9, 4 – 11, 4 – 13 list possible return values).

As to claim 14, Adaptec teaches to convey, by the adapter interface layer, the return data set into a format compatible with the application programming interface (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”); and translate the return data set into a format compatible with the application programming interface (Adaptec: page 3, Table 4 – 1 describes return values format, pages 5 – 18, Tables 4 – 3, 4 – 5, 4 – 7, 4 – 9, 4 – 11, 4 – 13 list possible return values). Day teaches that a translation layer translates return

data set into a format compatible with the application programming interface (Day: Figure 2, page 3, col 3, lines 10 – 25).

As to claim 15, Adaptec teaches to convey the return data set to an application that originated the received input/output request (Adaptec: page 1, section [Introduction] “ASPI allows developers to write device drivers and applications which are independent of the specific host adapter platform used to connect SCSI peripherals. This specification defines the protocol between ASPI client applications and the ASPI Manager”, page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”, page 3, Table 4 – 1 describes return values format, pages 5 – 18, Tables 4 – 3, 4 – 5, 4 – 7, 4 – 9, 4 – 11, 4 – 13 list possible return values).

With respect to claim 16, Raghunandan teaches a computer implemented system that receives an input/output requests and generates an input/output request formatted in accordance with a prescribed format.

Raghunandan does not explicitly teach that input/output requests formatted in accordance with an adapter interface layer as claimed.

Adaptec teaches to generate an input/output request formatted according to the claimed adapter interface layer (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”).

Raghunandan and Adaptec do not explicitly teach that a computer system comprises a processing element, a host adapter and a local interface communicatively

Art Unit: 2115

coupling the processing element and the host adapter to process an input/output request as claimed.

Day teaches the claimed computer system elements of a processing element (Day: Figure 1, element 102), a host adapter (Day: Figure 1, elements 124 and 100), and a local interface communicatively coupling the processing element and the host adapter (Day: Figure 1, elements 150 and 152).

Raghunandan , Adaptec and Day are analogous because they all teach generating and submitting input/output requests.

With respect to claim 16, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Raghunandan, Adaptec and Day. The combination of teachings would allow disk interface connections and protocols to be more flexible selected when integrating a personal computer or workstation motherboard with a storage controller.

As to claim 17, Adaptec teaches that the system wherein the adapter interface layer conveys the generated input/output request to the host adapter (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about installed SCSI host adapters and devices”, where SCSI host adapters and devices is a type of host adapter).

As to claim 18, Day teaches that the system further comprising a peripheral device communicatively coupled with the local interface via the host adapter (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about installed SCSI host adapters and devices”, where

Art Unit: 2115

SCSI host adapters and devices is a type of host adapter, and Day: Figure 3, elements 130, 300, 124, col 8, lines 13 – 32).

As to claim 19, Adaptec teaches that the system wherein the peripheral device provides a return data set formatted in accordance with the adapter interface layer to the host adapter in response to receipt of the generated input/output request (Adaptec: page 2, section [Using ASPI for Win 32 – An Overview], “ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests”); and translate the return data set into a format compatible with the application programming interface (Adaptec: page 3, Table 4 – 1 describes return values format, pages 5 – 18, Tables 4 – 3, 4 – 5, 4 – 7, 4 – 9, 4 – 11, 4 – 13 list possible return values).

As to claim 20, Adaptec teaches that the adapter interface layer is operable to receive a return data set formatted in accordance with the adapter interface layer from the host adapter in response to processing of the generated input/output request (Adaptec: page 1, section [Introduction], page 2, section [Using ASPI for Win 32 – An Overview], page 3, Table 4 – 1 describes return values format). Day teaches that a translation layer be used in the system (Day: Figure 2, page 3, col 3, lines 10 – 25).

As to claim 21, Day teaches that the translation layer translates the return data set into a format compatible with the application programming interface (Day: Figure 2, page 3, col 3, lines 10 – 25).

As to claim 22, Adaptec teaches that the system further comprising an application operable to generate the received input/output request, the application

Art Unit: 2115

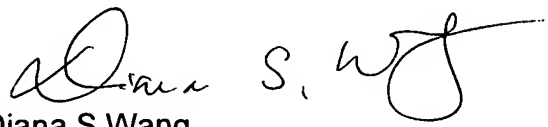
operable to receive a return data set formatted in accordance with the application programming interface (Adaptec: page 1, section [Introduction] "ASPI allows developers to write device drivers and applications which are independent of the specific host adapter platform used to connect SCSI peripherals. This specification defines the protocol between ASPI client applications and the ASPI Manager", page 2, section [Using ASPI for Win 32 – An Overview], "ASPI function calls can be used to retrieve information about the installed SCSI host adapters and devices, and to execute SCSI I/O requests", page 3, Table 4 – 1 describes return values format, pages 5 – 18, Tables 4 – 3, 4 – 5, 4 – 7, 4 – 9, 4 – 11, 4 – 13 list possible return values, Day: Figure 2, page 3, col 3, lines 10 – 25).

Contact Information


Any inquiry concerning this communication or earlier communications from the examiner should be directed to Diana S. Wang whose telephone number is 571-272-6522. The examiner can normally be reached on 8-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Thomas C. Lee can be reached on 571-272-3667. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).


Diana S Wang
Examiner
Art Unit 2115

January 10, 2006


SHAHID ALAM
PRIMARY EXAMINER